

Cloud Computing

Winter Term 2023/2024

Tutorial Session 2



Dominik Scheinert

Distributed and Operating Systems (DOS)

dominik.scheinert@tu-berlin.de

Assignment 1

- Submissions from 22 / 23 groups
- Reviews and gradings by 04.12.2023 at the latest
 - Once published, write me in case of questions

First Reminder

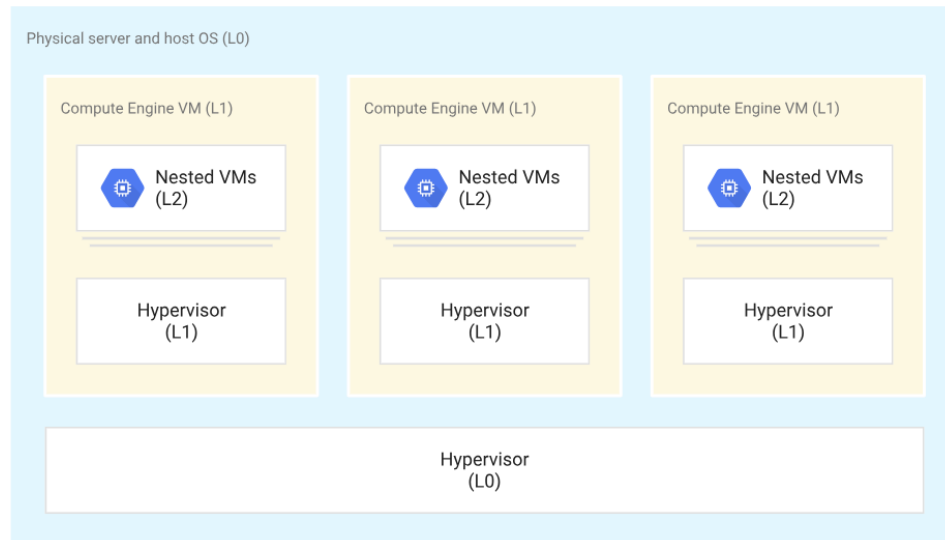
Always remember to shut
down your unused VMs!

Assignment 2

- In assignment 1, you were required to work with cloud platforms
 - From user perspective
- For assignment 2: switch role from cloud user to cloud provider!
- Task: deploy a cloud platform, make it usable for potential users
- As before: GCP VMs with nested virtualization are used as hosts

Nested Virtualization

- GCP VMs need to support nested virtualization!
- Investigate the possibility to enable nested virtualization
- Which virtualization type supports nested virtualization?

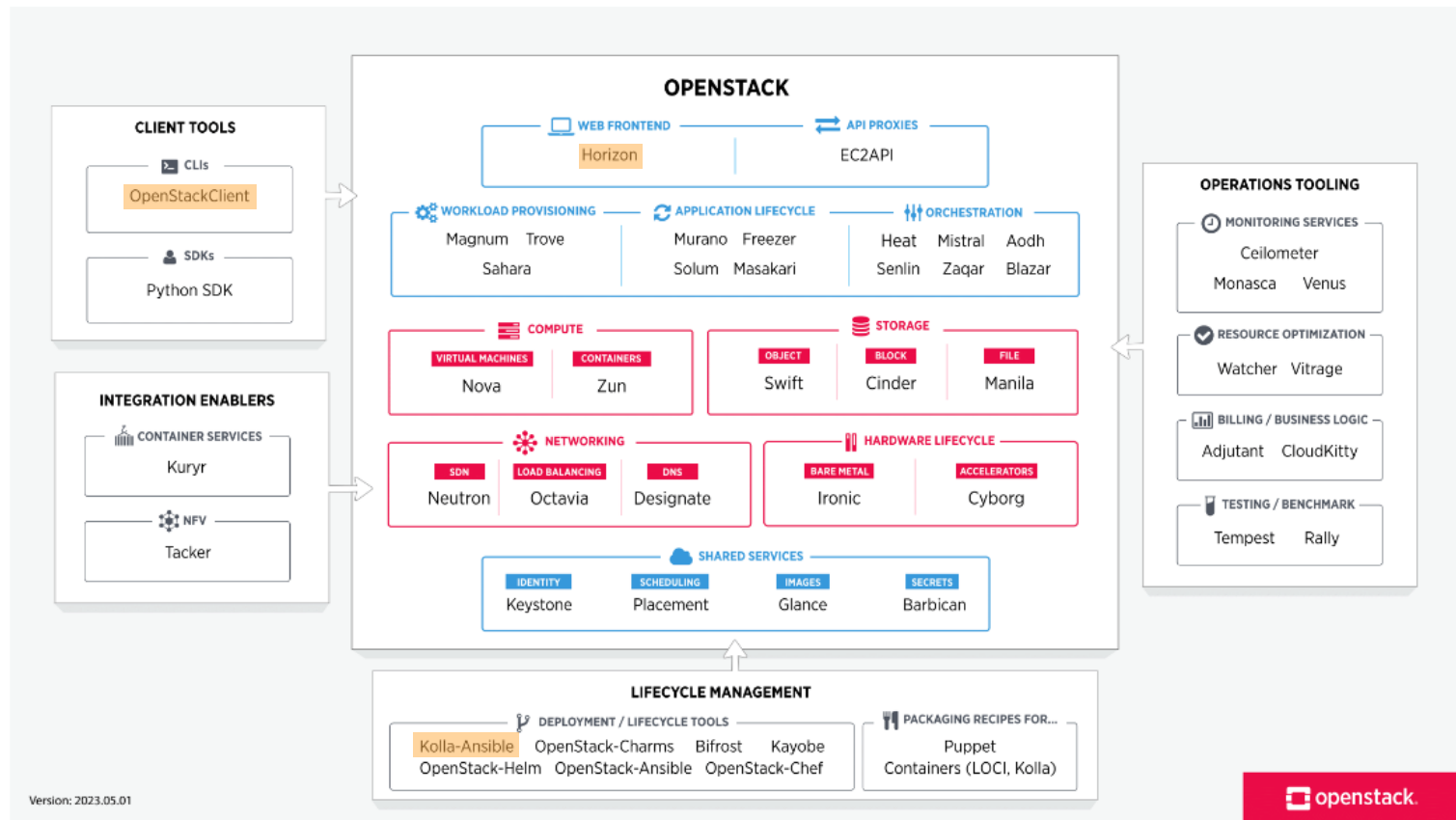


OpenStack

- Free open standard cloud computing platform
- Made up of many components for:
 - Computing
 - Storage
 - Networking
 - Identity management
 - Orchestration
 - ...
- Complex solution that easily exceeds scope of any assignment
- Simplified approach: we will deploy it using **kolla-ansible**



OpenStack



Kolla-ansible

- Open source community project since 2014
- Containerized deployment of OpenStack services
- Aims at simplification of OpenStack deployment
- Realized with Infrastructure as Code paradigm
 - --> Ansible



Infrastructure as Code (IaC)

- Apply software development techniques to infrastructure deployment & configuration
- Target infrastructure is any part of the organizational structure that is not the software itself
 - Servers, network nodes, storage devices
- Automation of installation & configuration
- Advantages:
 - Modularity, Reproducibility, Automatability, Testability, Scalability

Types of IaC

- Categories:
 - Imperative: Specify what to do
 - Declarative: Describe what you want
- Serving the configuration
 - "Push" changes to target infrastructure
 - "Pull" configuration from a central repository



Ansible



- Published in 2012
- Deployment & configuration described in YAML files
- Agentless -> Push
 - Only OpenSSH (and Python) needed
- Playbooks -> leans towards a declarative style
 - Contain configuration, variables and templates of a deployment
 - Using roles
- Roles
 - Describe a configuration of a specific component
 - e.g. database server; set up a VPN

Practical Assignment 2

- Task 1: Prepare GCP VMs for OpenStack
 - Use gained experiences with the Google Cloud Platform
 - Create a shell script that sets up a virtual machine environment
 - Disk image, VPC networks, VMs, firewall rules
 - VMs must support nested virtualization
- Task 2: Prepare deployment node and install OpenStack
 - Setup a python virtual environment (tested with Python 3.9)
 - Install dependencies in the provided requirements.txt
 - OpenStack deployment steps are provided
 - Adapt "multinode" and "globals.yml" files (check placeholders)

```
globals.yml
~/cache/fr-nzmr9p
Open Save
44 # Do not override this unless you know what you are doing.
45 #openstack_release: "2023.1"
46
47 # Docker image tag used by default.
48 #openstack_tag: "{{ openstack_release ~ openstack_tag_suffix }}"
49
50 # Suffix applied to openstack_release to generate openstack_tag.
51 #openstack_tag_suffix: ""
52
53 # Location of configuration overrides
54 #node_custom_config: "{{ node_config }}/config"
55
56 # This should be a VIP, an unused IP on your network that will float between
57 # the hosts running keepalived for high-availability. If you want to run an
58 # All-In-One without haproxy and keepalived, you can set enable_haproxy to no
59 # in "OpenStack options" section, and set this value to the IP of your
60 # 'network_interface' as set in the Networking section below.
61 kolla_internal_vip_address: "*****PUBLIC_IP_OF_CONTROLLER_VM*****"
62
63 # This is the DNS name that maps to the kolla_internal_vip_address VIP. By
64 # default it is the same as kolla_internal_vip_address.
```

YAML Tab Width: 8 Ln 154, Col 1 INS

```
multinode
~/cache/fr-daNrxm
Open Save
1 [all:vars]
2 ansible_python_interpreter=/usr/bin/python3
3 ansible_user=*****REMOTE_ANSIBLE_USER*****
4 ansible_become=True
5 ansible_private_key_file=*****PRIVATE_KEY_FILE_FOR_SSH_GCP_MACHINES*****
6
7 # These initial groups are the only groups required to be modified. The
8 # additional groups are for more control of the environment.
9 [control]
10 # These hostname must be resolvable from your deployment host
11 [*****PUBLIC_IP_OF_CONTROLLER_VM*****]
12
13 # The above can also be specified as follows:
14 #control[01:03] ansible_user=kolla
15
16 # The network nodes are where your l3-agent and loadbalancers will run
17 # This can be the same as a host in the control group
18 [network:children]
19 control
20
21 [compute]
22 [*****PUBLIC_IP_OF_COMPUTE1_VM*****]
23 [*****PUBLIC_IP_OF_COMPUTE2_VM*****]
24
```

Plain Text Tab Width: 8 Ln 20, Col 1 INS

Practical Assignment 2

- Task 3: Configure OpenStack
 - Use the provided scripts to
 - create the networks and
 - upload OS images (we use old ones for the sake of simplicity)
 - Write a script that prepares an OpenStack VM
 - Adapt NAT configuration on the GC controller VM to reach the OpenStack VM
- Task 4: Distributed Data Processing Frameworks
 - Create 3 Openstack VMs and prepare them with Java und Spark
 - Form a Spark cluster and verify connectivity
 - Submit example application and collect log files

Common Pitfalls & General Remarks

- Firewall Rules, Firewall Rules, Firewall Rules!
- Execute commands one after another; validate before proceeding
- Execute JavaSparkPi job from Spark examples
- Location of Spark logs depends on your chosen installation path

Assignment Submission: ISIS

- Submit all required files on ISIS
- Resubmissions are possible: only the last submission will be counted
- Submission will be partially validated automatically:
 - Use exactly the correct file names
 - Submit exactly the 9 required files described in the assignment
 - Text with answers to questions in submission text field / separate file

Last Reminder

Always remember to shut
down your unused VMs!